## Readlink

Value written in the output buffer is not null-terminated and may not contain the entire file name

Sean Barnum, Cigital, Inc. [vita<sup>1</sup>]

Copyright © 2007 Cigital, Inc.

2007-04-02

## Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 3903 bytes

Attack Category	Path spoofing	<ul> <li>Path spoofing or confusion problem</li> </ul>		
<b>Vulnerability Category</b>	No Null Termination			
	Indetermina	Indeterminate File/Path		
Software Context	Filename Management			
Location				
Description	of the file pointed. The value written terminated. Also, if the return input buffer, then	The readlink() function attempts to get the filename of the file pointed to by the given link.  The value written in the output buffer is not null-terminated.  Also, if the return value is the same as the size of the input buffer, then it is possible that the buffer does not contain the entire file name.		
APIs	<b>Function Name</b>	Commo	ents	
	readlink			
Method of Attack	output buffer may The filename is to buffer is smaller a Also, the filenam not a vulnerability of functions such overflows (due to unexpected result	The issue here is that the value written into the output buffer may not be the entire filename.  The filename is truncated if the length of the output buffer is smaller than the length of the filename.  Also, the filename is not null-terminated. This is not a vulnerability in itself, but subsequent use of functions such as strcpy() may cause buffer overflows (due to a lack of null-termination). Also, unexpected results could occur if the filename is truncated, and the programmer does not expect it.		
Exception Criteria				
Solutions	Solution Applicability	Solution Description	Solution Efficacy	
	Generally applicable to any readlink.	Always check the return value of readlink(). If the return value is equal to the length of the buffer, then	Effective.	

<sup>1.</sup> http://buildsecurityin.us-cert.gov/bsi/about\_us/authors/35-BSI.html (Barnum, Sean)

Readlink 1

Signature Details	make the buffer larger and retry.  Generally Always nullapplicable to terminate the any readlink.  int readlink(const char *filename, char *buffer,
	size_t size);
Examples of Incorrect Code	<pre>char buffer[100]; readlink (filename, buffer, 100); printf("The file name is: %s\n", buffer);</pre>
Examples of Corrected Code	<pre>char *readlink_malloc (const char   *filename) {   int size = 100;   while (1) {     char *buffer = (char *) malloc     (size);   int nchars = readlink (filename,     buffer, size);   if (nchars &lt; 0)     return NULL;   if (nchars &lt; size) {     buffer[nchars] = '\0';     return buffer;   }   free (buffer);   size *= 2; }</pre>
Source References	ITS4 Source Code Vulnerability Scanning Tool
	<ul> <li>http://www.gnu.org/software/libc/manual/ html_node/Symbolic-Links.html</li> </ul>
Recommended Resource	
Discriminant Set	Operating System • UNIX (All)
	Languages  • C • C++

## Cigital, Inc. Copyright

Copyright © Cigital, Inc. 2005-2007. Cigital retains copyrights to this material.

Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

For information regarding external or commercial use of copyrighted materials owned by Cigital, including information about "Fair Use," contact Cigital at copyright@cigital.com<sup>1</sup>.

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.

Readlink 3

<sup>1.</sup> mailto:copyright@cigital.com